

Zone-based Manufacturing

A Thesis Presented for the
Master of Science
Degree

The University of Tennessee, Knoxville

Abhay Rajendra Bajpai

May 2020

© by Abhay Rajendra Bajpai, 2020
All Rights Reserved.

Dedicated to grandfather-like figure, Madhavan Nampoothiri, who, although no longer with us, taught me how to live a content life without fear, for fear is nothing but something not known.

Acknowledgments

I would like to thank my parents, Anita and Rajendra, to have given their everything to get me where I am, for being supportive, both emotionally and financially, during this journey.

I would like to thank Dr. Rupy Sawhney to have put faith in my competence and ability to work on the development of this thesis and the completion of projects.

I would like to thank my committee members, Dr. John Kobza and Dr. Lee Martin, for taking their precious time out for my defense and provide invaluable insight into my thesis.

I am indebted to Dr. Ninad Pradhan for his undying patience and passion to help students achieve their objectives in research. He has been pivotal in directing my research. He not only brought hope with unique ideas and structure to the research but also improved my technical writing skills. Jokingly, he has been one of the founding members of a company called “Abhay’s Research & Effective Writing Skills Ltd.”

I would like to thank Dr. Roshanak Akram for being the sweetest friend and for her generosity in giving her precious time during her Ph.D. program in brainstorming and research development sessions of my thesis.

I would like to thank another founder of Abhay’s Research & Effective Writing Skills Ltd., Dr. Enrique Macias De Anda, for being my friend and giving me the motivational push for when I needed the most during my journey as a Master’s student.

I would like to thank Sara Keel for her stable and reasonable personality in helping me stay sane throughout my time of writing. I would also like to thank her parents for being my cheerleaders.

Thank you Vaikrant Pandav, Kshitij Sharma, and Prashanth Balasubramanian for being a great team for the success of the project which led to the inception of my thesis.

Abstract

Any improvements relating to productivity in an industry is a result of a decrease in the cycle time of the ongoing process. Traditional methods in a manufacturing industry focus only on increasing productivity by removing wastes from the process. The difficulty lies further in improving productivity once the wastes are removed. This could come at the price of an unequal distribution of work in people-based processes. The presented thesis emphasizes improving the production performance making sure that employees have an equal workload. The methodology develops a heuristic model to systematically produce a schedule of a manufacturing process with an increase in productivity. This schedule is made using real-world constraints, to have the least possible cycle time for the process for any given workstation in an assembly line. Furthermore, the difference between the duration of work of each employee is minimized. An optimization model performs an unbiased employee assignment mixed-integer linear programming problem. The results are proven by the case study provided.

Table of Contents

1	Introduction	1
1.1	Background	1
1.2	Problem statement	2
1.3	Approach	2
1.4	Thesis Organization	3
2	Literature Review	5
2.1	Overview	5
2.2	Efforts in increasing production performance	6
2.2.1	Cycle time reduction	6
2.2.2	Job Scheduling	9
2.2.3	Precedence & Overlaps	11
2.2.4	Heuristics-based approaches in scheduling	12
2.2.5	Integer programming-based approaches in scheduling	13
2.3	Spatial thinking	14
2.4	Efforts in workload balance	14
2.4.1	Integer programming-based approaches in workload balancing	16
2.5	Research gap	17
3	Methodology	19
3.1	Framework	19
3.2	Key definitions	20
3.3	Key assumptions	21

3.4	Precedence-based scheduling	22
3.4.1	Data collection	22
3.4.2	Management input	23
3.4.3	Precedence table	23
3.4.4	Precedence scheduling	25
3.5	Zone-based scheduling	26
3.5.1	Overlapping task	26
3.5.2	Precedence adjustments	28
3.5.3	Zoning and employee assignment	29
3.6	Workload balancing optimization	30
4	Case study	32
4.1	About Company ABC	32
4.2	Details of the facility	33
4.3	Floor-jig case	34
4.3.1	Precedence-based scheduling	34
4.3.2	Zone-based scheduling	40
4.3.3	Workload balancing optimization	42
4.4	Job-Task Analysis-II development	44
4.4.1	Job-Task Analysis by Company ABC	44
4.4.2	Job-Task Analysis-II	45
5	Discussion and future work	46
5.1	Concept of super-zones	46
5.1.1	What is a super-zone?	46
5.1.2	Benefits of super-zone	47
5.1.3	Inclusion for algorithm in super-zone	48
5.2	Other suggestions	48
	Bibliography	50
	Vita	55

List of Tables

2.1	Literature Comparison	7
3.1	Jobs i on the axis, where $i = 1:5$	24
4.1	Guideline to collect time study data	35
4.2	Employee assignment before ZBM	38
4.3	Floor-jig case of input	39
4.4	Floor-jig case: Results of zone-based scheduling and employee assignment. K denotes the number of assigned employees	42
4.5	Floor-jig case results of the employee assigned to minimize the variation of workload	43

List of Figures

2.1	The core concepts of ZBM	17
3.1	The framework of Zone-based manufacturing	20
3.2	Tasks at $t = 0$ units before alignment on a GC	25
3.3	Example of an incorrect overlap rule violation inferred using a task-based perspective	27
3.4	Example of a merger between overlap periods. The dotted and solid rectangular outlines indicate overlap periods. The hatched area indicates the merged overlap period, in which jobs A,B, and C are active.	28
4.1	The layout of the facility at Company ABC's facility	33
4.2	Recordings of Floor Jig through multiple angles	34
4.3	Snapshot of PMC which shows employee engagement at workstations for the case discussed	36
4.4	An example of a Gantt Chart including 5 employees	37
4.5	Floor-jig case - Gantt Chart of tasks at Floor Jig before application of ZBM	38
4.6	Floor-jig case - output of precedence scheduling. The resultant cycle time is 60 minutes.	40
4.7	Floor-jig case - output of Zone-based scheduling. The resultant cycle time is 39.33 minutes.	41
4.8	Employee time contribution ($K = 3$)	43
4.9	Job-Task Analysis of three employees and their respective responsibilities	44
4.10	Revised Job-Task Analysis of three employees and their respective responsibilities	45

5.1	Employee time contribution ($K = 3$)	47
-----	--	----

Chapter 1

Introduction

1.1 Background

Studies say this may cause an estimated 2.4 million unfilled positions in manufacturing industries between 2018 and 2028 [10]. To gain a competitive advantage, production companies try to increase production and achieve timely deliveries to their customers. This leads to the increased demand for physical work from employees at manufacturing facilities [8]. Increased demand for physical work results in a heavy workload because of the prolonged working hours of employees [31].

This workload may cause a rise in ambulatory activities especially at an assembly line with a large work-space [33]. However, an increase in such activities does not add value to the production [30].

‘Cycle time reduction’ is identified as the most desirable outcome to increase production performance [23] [15]. Solutions provided by techniques like Continuous Improvement and Lean Thinking reduce activities that are not valuable to the operation, also called “non-value-added activities”. This reduces the cycle time or increases throughput as the non-value added waste is removed from the process [25]. On the removal of non-value-added activities, it becomes challenging to achieve a further reduction in cycle time as only value-added activities, which are necessary and non-expendable, remain in the process.

Cycle time reduction does not guarantee an equal workload for all employees as one employee may be assigned more value-added activities relative to others. Hence, the

assignment of employees to these value-added activities, while balancing their workload, adds a layer of complexity to the already existing challenge of cycle time reduction. An increased workload to meet high production expectations exacerbates the risk of unbalanced workload which in turn hurts the production performance [22]. Hence, there should be an approach which not only meets production demands but also avoids negative effects on the employees' well-being through an equitable workload.

1.2 Problem statement

The problem at hand is to improve the production performance while balancing the workload of the employees, especially at facilities having unequal employee workload. The following research questions motivated the development of this problem:

1. Is it possible to reduce the cycle time by focusing on value-added activities of a manufacturing process at a workstation?
2. Can activities be divided into different zones at large-sized manufacturing facilities?
3. Can value-added activities be shared among the employees assigned to the workstation?
4. If sharing is possible, how can we achieve an equitable workload distribution?

1.3 Approach

Zone-based Manufacturing (ZBM) is the approach proposed for reduction in cycle time with a balanced employee workload. It is applied to manufacturing processes at a workstation of an assembly line. ZBM provides the following outcomes: 1. reduction in cycle time by the sharing of tasks (value-added activities) among cross-functional teams using zones, 2. generation of a schedule of tasks using a heuristic approach, and 3. distribution of equal workload using a linear programming model. The following is an overview of the steps in the ZBM methodology:

1. **Precedence-based scheduling (PBS):** 1.) Collect data related to the time taken by each task to complete and the number of employees assigned to each task, 2.) Develop constraints based on the observed sequence of the process, and 3.) Schedule and align tasks based on these observations. Following these steps, PBS generates a schedule of the process and marks the initial step towards cycle time reduction.
2. **Zone-based scheduling (ZBS):** ZBS achieves a further reduction in cycle time in the schedule generated by PBS based on: 1.) the concept of “overlap”, and 2.) availability of employees. An “overlap” is a condition when multiple tasks assigned to a workstation are performed in parallel and shared among employees. The physical area of a workstation at which an employee performs their share of the team task is referred to as a “zone” in this thesis.

The final schedule generated from ZBS contains the final sequence. The sequence of tasks and reduction in cycle time is visualized using a Gantt Chart (GC).

3. **Workload balancing** assigns all employees to the final schedule of the process at the workstation based on a binary-integer linear programming model. The objective function used is a goal-chasing function to achieve a balanced workload for all employees assigned at the workstation.

The ZBM was piloted at multiple workstations in a facility of Company ABC. The results of these pilot studies include a final schedule for the process which demonstrates a reduction in cycle time at the workstation and a balanced employee workload.

1.4 Thesis Organization

This thesis is organized in five chapters. Following this introductory chapter, Chapter 2 provides a literature review of articles and analysis of the research gap relative to the study of cycle time reduction, scheduling processes to increase the performance of production, balancing of workload shared by the employees, spatial thinking for cycle time reduction, and techniques used for finding solutions through algorithms. Chapter 3 dives into the formulation of a model that uses the concepts and techniques discussed in Chapter 2.

Chapter 4 provides the background information of the facility used for the application of the methodology, and is used for validation purposes. Chapter 5 discusses the scope of this research with certain improvements to include how this methodology could be used if these improvements are made.

Chapter 2

Literature Review

Improving production performance by reducing cycle time is one of the sought-after objectives of manufacturing industries. Research published in production and industrial engineering journals provides a plethora of options to achieve improvements in productivity. Adjusting the operations/tasks in a schedule to obtain the least operating time under given circumstances increases productivity. This thesis adopts the definition of “schedule” from [7], as ‘the time intervals assigned to operations/tasks to be performed by machines or employees’. However, a machine-based manufacturing environment does not represent the entire manufacturing sector. Several industries, with a predominant people-based manufacturing environment, increase their productivity by appropriately using their workforce [33] [4].

Google Scholar, Web of Science, Scopus, and University of Tennessee’s Library’s OneSearch engine were used for the literature survey:

2.1 Overview

This chapter provides a comparison of methods to improve production performance and a balanced workload for an employee. The literature review is based on 36 articles related to the improvement of productivity in industries like manufacturing, construction, shipbuilding, and healthcare. The literature is summarized in Table 2.1. The table also shows techniques chosen by the authors to solve the respective problems. Approximately 63% of the articles focused on the reduction of either lead time or cycle time to improve production performance.

Of these, 30% addressed workload distribution among the resources used. The literature is categorized for the rest of the chapter as follows:

1. Efforts in increasing production performance : This section highlights approaches to obtain the objective of increasing productivity through cycle time reduction. In a challenging manufacturing market, machines are used to meet production demands. Only 20% of the research, which focused on cycle time reduction, had partial or full involvement of human resources or employees in the process. Hence, this section shows research focused on objectives used in industries that rely more on machines than employees.
2. Spatial Thinking : This section provides an insight into how some work is performed or shared in a workspace that affects the cycle time of the process. Spatial thinking influenced the concept of zoning in this thesis.
3. Efforts in employee workload balancing : This section highlights the use of binary-integer linear programming methods for scheduling which result in reduced employee hiring costs. Some of these articles present solutions to a balanced employee workload distribution.

2.2 Efforts in increasing production performance

2.2.1 Cycle time reduction

Cycle time (CT) in scheduling problems is commonly referred to as “makespan” and has the same definition, i.e. the time required for completion of all the jobs required at a workstation. A considerable amount of scheduling-related research defines its objective functions based on makespan minimization. Around 45% of papers found in the research for flexible job-shop scheduling (FJSS) have makespan reduction as objective function. About 24% of the same articles define multiple objective functions which includes minimization of makespan, the workload of most loaded machines and the total workload of machines [6]. Cycle time reduction is one of the most common objective functions in optimization

Table 2.1: Literature Comparison

Papers	Concepts Used					Techniques Used		
	CT/M	SC	P/OV	WB	ST	H/MH	IP	LT
Ouelhadj and Petrovic (2008)	x							
Panwar and Jha (2019)	x	x					x	
Al-Araidah et al. (2010)	x			x				x
H Nie et al. (2014)	x	x	x	x		x		
M.S.Mehrabad and P. Fattahi (2007)		x				x		
Costa et al. (2013)	x					x	x	
Gupta and Stafford Jr (2006)		x						
Seung Heon et.al.	x							x
Bultmann et al. (2018)		x					x	
Rajendran and Ziegler, 2004	x	x				x		
Afzalirad and Rezaeian (2016)	x	x	x				x	
Chaudhry and Khan (2016)		x						
Calis and Bulkan (2015)		x						
Vallada and Ruiz (2011)		x						
D. Prot and Morineau (2017)		x	x					
Serpil Mutlu et al. (2015)		x					x	
Dolgui et al. (2018)			x			x	x	
Murat Firat, C.A.J. Hurkens (2012)		x					x	
Fattahi et al. (2009)		x	x			x	x	
Lee et al. (2012)		x	x				x	
Tse et al. (2004)	x				x			
Koh et al. (2011)			x		x			
Kwon, Lee (2015)	x	x	x	x	x	x	x	
Demir and Isleyen (2013)	x	x	x				x	
Ernst et al. (2004)							x	
Giglio et al. (2017)							x	
Moon et al. (2009)							x	
Pastor et al. (2002)				x		x		
Slomp et al., 2005				x			x	
N Azizi and M Liang (2012)				x			x	
R. B. Bachouch et al (2010)				x			x	
Olga Battaïa et al. (2015)				x		x	x	
Sirovetnukul, Chutima (2010)	x			x			x	
Bouajaja, Dridi (2016)				x				
Lian et al. (2018)				x			x	
Zone-based Manufacturing	x	x	x	x	x	x	x	x

Note: CT/M: Cycle Time/Makespan; SC: Scheduling; P/OV: Precedence/Overlap; ST: Spatial Thinking; H/MH: Heuristics/Metaheuristics; IP: Integer Programming; LT: Lean Thinking.

problems. Depending on the type of industry, CT/makespan optimization was performed in conjunction with either cost, resources, environmental impact, and others.

In research done by Panwar and Jha, CT reduction is one of the multiple objectives which also include optimization of the cost associated with materials, equipment, labor, and environmental impact [28]. Project duration is an important parameter of construction projects, and is determined. Reduction in the CT, which in this case is known as “total project duration”, happens by creating nodes of activities and reducing total duration to complete the project. A network of nodes was constructed, which represented the schedule of the entire project. Although they find the trade-off between CT reduction and other objectives, CT reduction was the primary objective.

Mehrabadi and Fattahi provide a job-shop manufacturing problem [34]. This research achieves CT reduction by choosing the right machine to accomplish the tasks and by organizing the tasks for the best sequence. Each job requires several operations and are dependent on a sequence. One job contains several operations. Hence, the complexity arises because of not performing all operations on just one machine.

Rajendra and Ziegler perform makespan reduction in flowshop manufacturing where machines process certain tasks [32]. The sequence of processing jobs on machines is identical since this is a permutation flowshop problem. Defining the start and the end time of each job helps in calculating the makespan and equations which are used to address the sequence of these jobs. The equations are used to calculate the processing time of the sequences for all machines, which is known as flowtime, which is minimized in this case.

Afzalirad and Rezaeian discuss unrelated and a parallel machine problem in which tasks line up to be performed on these machines [1]. Since the machines are unrelated, their utilization is planned using setup times which are dependent on a sequence. The processing times of these tasks are subject to constraints related to resources and are affected by their availability. The objective function is to reduce the makespan by timely assignment of these machines to the tasks.

In Lean methods reduction in waste or non-value added activities help industries in reducing CT. Lean methods like DMAIC and 5S were used by [2] in healthcare. DMAIC-based procedures were used to reduce the CT of the drug dispensing process by 45 percent.

In research done by H. Nei et al [27], the problem is defined for multi-skilled workers with different skill levels required to complete a project in a cellular manufacturing environment. The objective function of CT reduction is by completing these activities under given human resources and given time. By using scheduling and cost reduction of salaried employees, a final schedule is provided which satisfies the objectives under employee skill and availability constraints.

Research provided in [9] presents a problem for parallel machines and human resources assigned to tasks. The authors explain how a reduction in the setup time can have a positive impact on the overall CT reduction. This setup time is taken by the employees assigned to work on the machines to have these machines ready to produce the final products. By making employees available on time, the setup time on the machine is reduced. The selection of employees based on their skills, on how long they take on performing a setup is also considered in this problem.

The paper discussed by [33] shows ways to reduce cycle time from the perspective of the employees assigned to jobs instead of the perspective of reduction of the operating time of machines. The assignment of employees to the machines is followed by walking to multiple machines in a U-shaped assembly line. This walk-time is included in the objective of cycle time reduction in addition to the processing time of the tasks. The walk time is calculated by using straight-line distance and walk-speed.

2.2.2 Job Scheduling

Scheduling is chosen as the method by healthcare teams to synchronize their availability, which impacts timeliness of a procedure [26]. Apart from healthcare, the manufacturing industry also depends on sequence-related procedures. Unlike healthcare, the dependencies on teams of machines might not be high. Although, in cases where machines do not depend on each other, they do depend on the tasks they are scheduled to work on. This is commonly known as unrelated machine scheduling. Scheduling is a tool to solve sequence-related problems and its objective is to increase productivity. This is done by the ability of scheduling for timely completion of the required tasks under given sets of constraints.

In job-shop manufacturing, scheduling increases productivity by cycle time/makespan reduction [17] [6]. Flow-shop manufacturing also uses scheduling for cycle time reduction [14]. A survey article, on scheduling in flexible job-shop manufacturing, found that 47.6% of the articles include makespan minimization in their objective function [5]. The objective is achieved under several constraints like tasks dependent on the sequence, availability of resources, operational costs, etc. [12] resonates the most with this thesis as it states the importance of the sequence of the task, as well as the cycle time being affected by the start time of each job.

Apart from scheduling in job-shop, flexible job-shop and flow shop problems, scheduling is also used in construction problems. Various alternatives to complete the construction project are provided under an assumed time horizon [28]. All the resources like materials, equipment, and labor, required for various alternatives, are listed. The variation in the usage of these resources causes variation in the timely deliverance of the building. An optimized schedule is found with the most reliable timeline for a project. Hence, a many-objective scheduling model is constructed. A similar approach is provided by Hui & Bo in [27] in which a scheduling model, including multi-skilled employees, is developed. A schedule is provided for m multi-skilled workers with different skill levels to finish n tasks. The sets for skills and human resources are defined. These resources are later assigned to the n tasks. The problem is developed into an optimization problem by setting objective functions of makespan and cost reduction.

Parviz Fattahi approaches to solve complicated sequencing and scheduling of job-shop problems in [13]. The challenges lie in assigning operations to appropriate machines and sequencing the operations on each machine. The problem has n jobs and m machines and each job contains certain required operations. Each operation for a job needs to be scheduled on these machines. Cycle time of the entire process is calculated by the addition of all job completion time which is a function of processing times of each operation. This idea of task assignment based on reducing the processing time on each machine resonates the most with ZBM.

In a flowshop scheduling problem, discussed by Rajendra et al. in [32], contains several identical tasks on all machines hence making it a ‘permutation’ flowshop problem. Since

it is a flowshop problem, these tasks are sequence-dependent. In this problem, there are numerous possibilities of choosing a schedule and therefore optimizing the schedule becomes time-consuming. To resolve this, the researchers make a search algorithm to find a quick solution rather than finding the most optimized.

Afzalirad tried to schedule parallel machines that are also unrelated to each other [1]. This also becomes similar to the permutation flowshop as there could be numerous possibilities to perform scheduling. The researchers were interested in dealing with a problem based on constraints provided in a real-world situation. One real-world application of this type of problem is the erection of blocks in the shipbuilding processes where cranes or forklifts are unrelated but work towards making a ship with many blocks inside it. The processing time of each operation is defined along with its overall sequence. A solution was found using meta-heuristics which gave a near-optimal solution. This complicated problem is solved using a novel optimization model for release dates, sequence and machine-dependent setup times.

2.2.3 Precedence & Overlaps

Several scheduling-related research consider inter-dependencies of the task to define the sequential flow of the process. These inter-dependencies form the basis of the sequence and are commonly referred to as ‘Precedence’ or ‘Precedence relations’. The inclusion of precedence relations in scheduling makes it practical, as it resembles real-world problems.

Precedence relation in research done by H. Nie et al. is developed for nine tasks where the precedence becomes a constraint for each task’s initiation [27]. In [1], the precedence constraint does not allow a task to initiate unless the preceding task is done.

Demir et al. used precedence in the form of a binary variable where the value of it is 0 if a job assigned on the machine has no precedence over the successive job [11]. Similar to the application of precedence as a binary variable, [20] uses ‘AND’ and ‘OR’ to let two tasks exist in parallel and not allow two jobs to exist in parallel, respectively. In the U-shaped assembly line manufacturing problem [33], precedence was used for several tasks ranging from 7 to 297. They developed a way to find precedence relations that are concentrated at some places by using networks. A similar concept to precedence is ‘overlap’ and is rather uncommon to represent the inter-dependencies of the tasks. This is another way of representing the

allowance of two machines/employees to work in parallel [13]. These tasks may not be shared among employees/machines. Inspired by this work of overlapping model, [18] used the variable called overlapping coefficient where the value is equal to 1 if no overlapping is allowed.

2.2.4 Heuristics-based approaches in scheduling

Scheduling is a well known theoretical and practical problem that uses heuristics as a way to achieve high-quality solutions in a short amount of time. Heuristics come handy when the scheduling processes use hard combinatorial problems. A large variety of settings that mimics scheduling problems in the real world can be represented using heuristics. One of the surveys done in multi-objective scheduling problems finds that meta-heuristic approaches were used in 36 out of 53 articles mentioned [21]. These findings commonly used heuristics methods like Tabu Search (TS), Genetic Algorithm (GA), Simulated Annealing (SA), Cuckoo Search (CS) and Ant Colony Optimization (ACO) [32].

H. Nei et al present a CS heuristic which aims to handle conflict in scheduling multiple jobs according to the priorities for an industry with multi-skilled workers [27]. TS algorithm-based heuristic used by Mehrabad et al., is performed in two stages to find the best sequence of the jobs, as well as the best choice of machine selections [34]. The algorithm iteratively compares the current solution with the next and, better feasible solution to improve operation sequences.

Similar to TS, Ant colony algorithm (ACO) is another search algorithm used in scheduling problems. The algorithm is inspired by ant pheromone trails used by real ants as the medium of communication and feedback. This is usually done by creating a loop after initializing pheromone trails and parameters until a solution is constructed by the local search to update the pheromone trail to return a solution. [32] used ACO to find the best makespan for the permutation flowshop problem. They not only used the traditional ACO but also developed an improved version of ACO to solve their problem.

A few heuristic methods generate solutions based on a more complex algorithm like a Genetic algorithm (GA). GA is a heuristic method inspired by the process of natural evolution as it works with a set of population generations. Every iteration produces a new

population from the previous population with the help of “mutation” used in chromosomes. GA is used by Costa et al. in their research to find the order of jobs and workers assignment [9].

Another search algorithm focuses on the Simulated Annealing algorithm (SA) inspired by the metal cooling process to aim to get closer to an optimal solution with gradual temperature reduction. The initial solution is created randomly and only then is a neighborhood search begun. In research done by Khalife et al. [18], a loop is run until the temperature meets the optimal required temperature. This is performed for employee assignment and scheduling. They used SA to solve a multi-objective FJSS problem by reducing the makespan and total machine work loading time which in turn improved productivity.

2.2.5 Integer programming-based approaches in scheduling

Using linear programming is the most common practice of representing a variety of constraints and defining an objective mathematically. Mixed-integer linear programming (MILP), a subcategory of linear programming, is widely used for its ability to use both continuous and discrete values. Discrete and continuous values are used to represent units of resources and processing time of tasks respectively, in scheduling problems.

Al-Araidah et al. created a MILP model with a single objective function on the reduction of makespan of workstations with worker teams and machines [2]. Similar to research, numerous papers have their objective function as makespan/cycle time reduction. M. Afzalirad [1] shows work to reduce makespan in a parallel machine scheduling system with a few operational constraints while Demir et al. [11] present the same objective function in flexible job-shop manufacturing.

Chutima et al. take a different approach in which the researchers use MILP modeling to reduce the walk-time and workload of employees in a multi-objective problem [33]. Khalife et al. provide a multi-objective approach to minimize the makespan and machine loading time in a flexible job-shop environment [18]. The novelty is the flexibility in the weight given to each of these objectives for the user of this model for customization. In a construction scheduling research [28], the authors present minimization of a multi-objective model which

reduces total project duration along with the project cost, resource moment and CO_2 emissions.

2.3 Spatial thinking

“Spatial thinking” is a way of dividing tasks into several parts of a given work-space. An example of this can be explained in research conducted through the interaction of people and computers [36]. Participants, in teams of two, were provided one mouse each to operate on one shared computer screen. Participants were asked to perform a sketch using their mice. They spatially divided the sketch with an imaginary line at the center of the computer screen. Although sketching is incomparable to tasks required in a manufacturing environment, the idea behind sharing a task remains the same.

Another research in spatially dividing the work is in shipbuilding. This is discussed in [19] in which the problem considers assembling the sub-parts called “blocks” into a ship-hull. The already manufactured blocks are scheduled to be sent on the hull of the ship. The movement of these blocks is performed by unrelated parallel machines strategically placed in a large ship-hull. The idea behind this placement is to schedule and balance the workload among all the individual machines to attain the most output in a given time. This is spatial thinking in a nut-shell. In this paper, an orchestration of operations was performed with the due date as one of the constraints.

2.4 Efforts in workload balance

The utilization of resources in scheduling affects the completion time of the process because of the availability of the resources. In this section, efforts were made by researchers that provide insights on resource utilization along with balancing workload.

Research done in healthcare has been observed to balance the workload of nurses and doctors performing surgeries, drug delivery, planning activities, and other desk jobs. A study provided by Bachouch et al. shows an optimum assignment of tasks for different sets

of nurses while also considering their skills [3]. Tasks are not shared by nurses/doctors as healthcare relies on compartmentalized skills.

Workload discussed in the problem and in research is not uniform across different workstations [29]. The objective was to have an equal cycle time at all the workstations to achieve a uniform workload distribution. This was achieved by minimizing the random distribution of tasks to the line workers on different models using the sequencing of tasks as a constraint. Once, a workstation was balanced, a search algorithm was used to continue balancing the workload at other workstations. This guarantees a balanced workload for every worker. Battaïa et al/. visualized the demand at the assembly line as the workload of employees [4]. The workstations with a heavy workload, when a more strenuous operation is being performed, were assigned more employees than previously. The solution balances work at the assembly line and hence provides equal work distribution at each station. All workers at these workstations were assumed to be identical and were allowed to switch between workstations. Research done by Chutima resonates the most with the work done by this thesis [33]. The U-shaped assembly line contains employees who are required to walk to different workstations for a few tasks. Hence walk-time is included in their workload. The first objective function is to reduce the deviation of operation times of workers. The second objective function is specifically for the minimization of walk-time. A constraint is developed not to let an employee's workload exceed the cycle time of the workstation. Later the overall workload, at each workstation of the assembly line, is also achieved. This balances both, the assembly line and the workload of all employees assigned to each workstation.

The manufacturing problem considered by [24] is in cellular manufacturing in which workers differ from one another in terms of skill sets and proficiency levels. The availability to be assigned to multiple tasks is allowed under skill-related constraints. The mathematical model is developed to minimize the workload of the entire cell as an entity and to also balance the average workload of the employees. This paper does not discuss an individual worker's workload in a cellular manufacturing system. A study performed by Slomp et al. balances the workload based on the qualifications of employees [35]. The interesting approach in this paper was to find the bottleneck workers. In other words, workers which at any given point of time had the highest demand as a result of their cross-functionality. The model provides

a solution by training them to operate machines and making them available when needed on machines so that the workload of all workers is equal. However, task-rotation and sharing were not considered.

2.4.1 Integer programming-based approaches in workload balancing

Employee-assignment and workload balancing problems have different objectives and constraints from scheduling problems. This difference lies in the change in focus from balanced workstations to balanced work for individual employees. A multi-objective model used by Slomp et al. optimizes the requisite training of workers for their assignment on available machines so that employee absenteeism, as well as fluctuations in demand at the workstations, becomes manageable [35]. The constraints provided regard training costs and the multi-functionality of the employees. Maintenance and training costs are included in this model by assigning weights to the objective function.

Apart from the manufacturing sector, the model for employee assignment was also used in the healthcare sector to minimize the difference between the maximum and the minimum workload of each nurse working in a hospital [3]. While the objectives in healthcare might be different from the manufacturing sector, there are similarities like training cost and skill requirements exist in the usage of MILP. Another MILP model to balance employee workload discussed by Battaïa et al/ achieves assembly line balancing using a single objective function to minimize the number of workers used at a workstation [4]. Constraints mentioned here are to guarantee that task-sequencing is obeyed and that workers do not work on more than one task at any given period.

2.5 Research gap

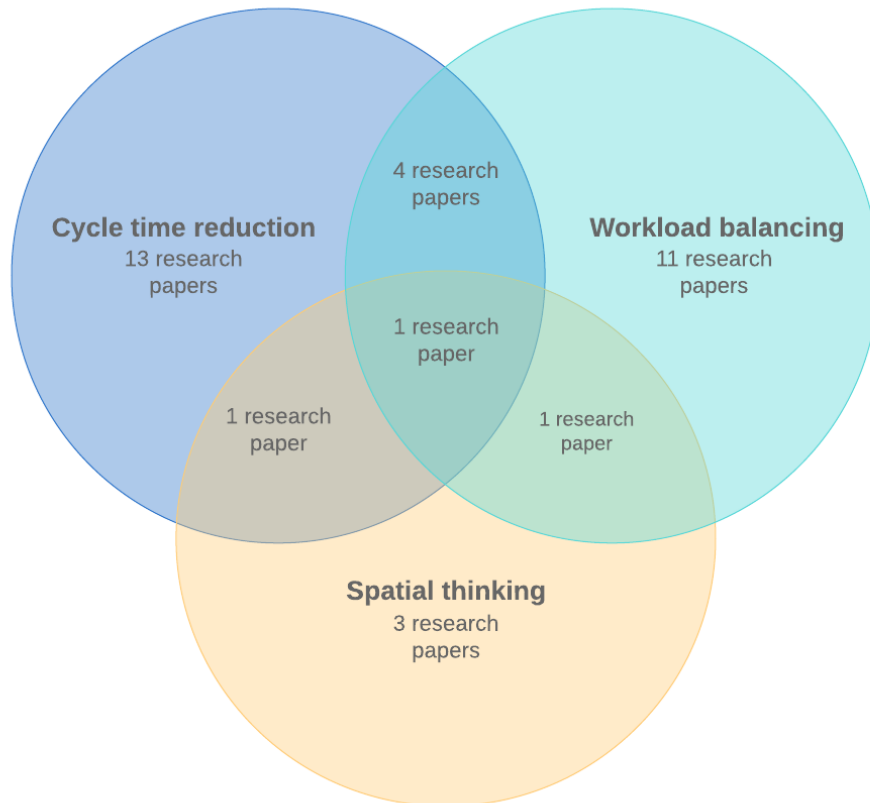


Figure 2.1: The core concepts of ZBM

Although this research provides insight on very specific problems, research that combines all the above-mentioned concepts in a single framework was not found. The methodology of ZBM considers the framework which includes CT/makespan reduction, workload balancing and spatial thinking as shown in figure 2.1. Although, there is abundant research in makespan reduction by reducing setup-time or by removal of non-value added activities. However, the reduction of makespan by scheduling only the value-added tasks is not explored.

Inter-dependencies of tasks or precedence in the scheduling problem is not considered in the flexible job-shop or flow shop approaches. Additionally, precedence was always

implemented in heuristics as it was always introduced as a constraint in linear programming models [28][18], [13].

In most of the research, spanning from parallel machine scheduling in the manufacturing sector to scheduling nurses in healthcare and the construction industry, task-sharing among employees or resources is not considered. In the manufacturing sector, workstations have only one employee, leaving no possibility for task-sharing. A majority of the scheduling problems in the same sector do not consider sharing tasks required at the workstations [27] [3] [11] [33]. This is typical in both machine and employee-based systems [1] [9]. Even the research closest to this thesis [19] does not consider sharing tasks as the machines are unrelated. In research done in the construction industry, task completion was given priority rather than sharing of tasks using employee assignment. In healthcare, sharing of tasks was not observed as nurses and doctors were not allowed to do so.

The majority of research in scheduling does not consider the physical space of the manufacturing area. In most of the research mentioned, employees work at small-sized manufacturing workstations which allows little to no overlapping. This thesis considers an environment where much of the work is completed by human employees who share tasks with little automation in a large manufacturing environment. ZBM does not change the nature of the tasks at each workstation but instead focuses on work sharing and assignment to ensure a balanced distribution of work.

A review of the number of researches performed in balancing the workload of the resources is provided by [24]. Only 1 out of 23 listed citations, had an objective function of workload balancing in a cellular manufacturing system.

Chapter 3

Methodology

3.1 Framework

A flow chart of the framework of Zone-based Manufacturing (ZBM) is shown in figure [3.1](#). ZBM uses three main steps, 1.) PBS, 2.) ZBS, and 3.) Workload Balancing to achieve CT reduction and balanced employee workload. These objectives require the collection of the following data: 1.) sequence or relative position of a task in comparison to other tasks, 2.) duration of each task and the number of employees performing it, and 3.) total cycle time of the process.

After data collection, PBS aligns all tasks in a sequence using scheduling. Scheduling in PBS marks as an initial step towards a reduction in CT purely based on the chronology and relative-positioning of tasks. In the following step, ZBS further reduces the CT of the process and assigns an appropriate number of employees to each task at the workstation based on the availability of each employee. ZBS achieves this by using the concept of ‘zones’ which allows multiple employees to share a task.

Based on the final schedule obtained after performing PBS and ZBS, the next step assigns employees to specific tasks in the schedule. This step is referred to as ‘Workload Balancing’ where the workload of each employee is evaluated to achieve the objective of equal distribution of workload.

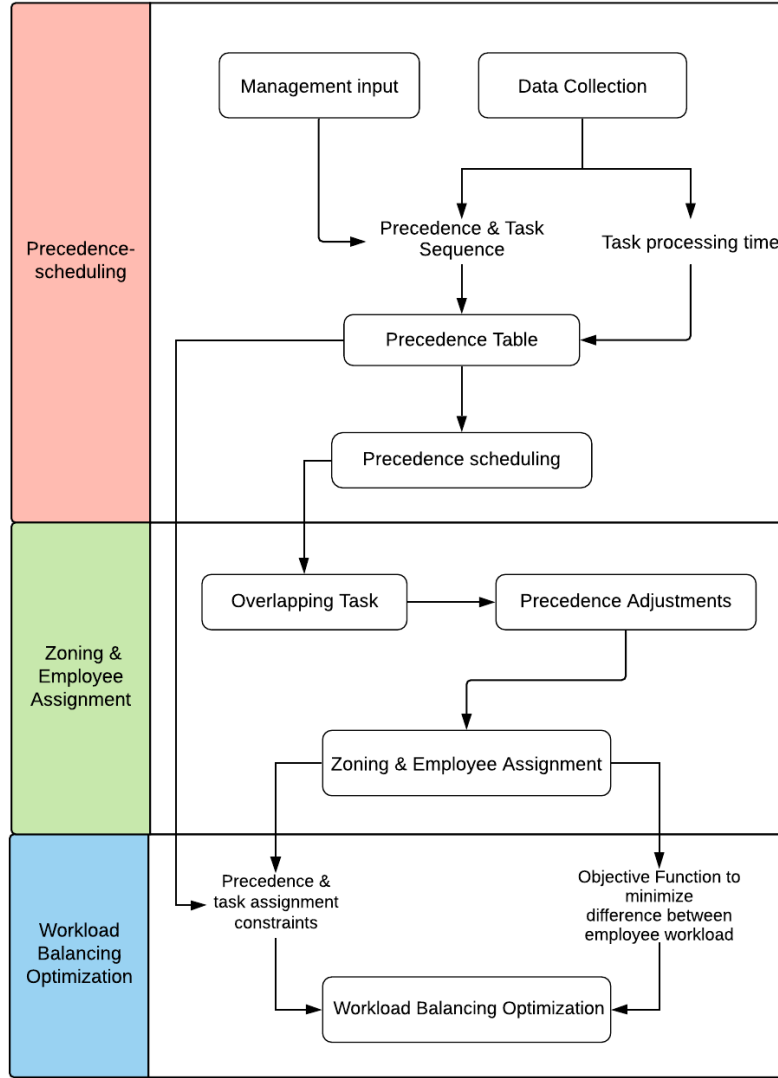


Figure 3.1: The framework of Zone-based manufacturing

3.2 Key definitions

Although few definitions were previously introduced by other researchers, they are used in the ZBM to achieve a different objective. The following are the definitions that are used in the rest of the thesis:

- **Zone:** A physical work-space in which a cross-functional employee performs part of a shared task.

- **Overlap:** A condition when two or more tasks are active at the time or exist in parallel.
- **Overlap Period:** The time for which two or more tasks have an overlap
- **Distributive Task:** A task having any kind of polarization, clustering or regional concentration across the workstation. For example, the task of plumbing in a house is concentrated around the bathrooms and the kitchen. Hence, plumbing is a distributive task.
- **Task processing time:** The difference between the start and the end time of a task.
- **Start/End time:** The unit of time on the Gantt chart's horizontal axis when a task begins/finishes.
- **Precedence:** A condition where if task A has precedence over task B, task B is not allowed to work in parallel to task A. Task B can only begin after task A is finished.
- **Precedence factor:** A value assigned to a task that defines its precedence over other tasks.
- **Task-Sequence:** The flow of the process in which tasks are lined up in chronological order which obeys precedence.
- **Active-task:** A task that exists during an overlap period.
- **Employee assignment :** The act of scheduling an employee to work on a particular task(s).
- **Shared task:** A task that has more than one employee assigned to it.

3.3 Key assumptions

Non-value-added time is not included. During data collection, Person Machine Chart (PMC) labels a task as either 'non-value added' or 'value-added'. Non-value added tasks are filtered and removed when task processing times are registered because ZBM focuses on CT reduction using value-added tasks present at the workstation.

All value-added tasks are assumed to be capable of being ‘zoned’. Since ZBM focuses on manufacturing areas with a large footprint, the division of a single job into multiple zones is feasible.

The task-sequence of ZBM only allows full precedence of a task over another task. This means if task A has precedence over task B, task B can only begin after task A is complete.

All necessary equipment and materials are provided to the employees assigned at that workstation. This assumption removes non value-added time spent by employees while they wait for the equipment to be handed to them by a previous user.

Multiple employees can be assigned to the same task. This assumption is based on the previous assumption of material and equipment availability, which makes it possible for each employee to work without a waiting period to share resources.

Task processing time has a negative linear relationship with the number of employees assigned to it. Therefore, having multiple employees work on a task linearly reduces the task processing of that task.

The employees are assumed to be cross-functional. This implies that an employee may be assigned to any task at the workstation. This assumption may be relaxed in the future by introducing the concept of a ‘super-zone’, discussed in Chapter 5.

3.4 Precedence-based scheduling

3.4.1 Data collection

One of the objectives of data collection is to record the task processing time for all tasks when performed by only one employee. In practical settings, tasks may have multiple employees assigned to them. Hence, task processing times of such tasks would not be the same as the task processing time if a task was performed by one employee. Task processing times of tasks with multiple employees were altered using equation 3.1. If a task has one employee assigned to it, the task processing time is unaltered.

Let t_{jk} be the task processing-time of task j and k employees performing that task. The alteration is performed by multiplying k employees to the task processing time recorded for the task j as shown in equation 3.1. The multiplication gives t_j , which is the altered task processing time of the task j . If there are multiple unique values of the task processing time, the highest value is accepted as the task processing time for that task.

$$k \times t_{jk} = t_j \quad (3.1)$$

The task-sequence and task processing time, together provide the following insights: 1.) the cycle time of the entire process and, 2.) the relative position of a task as compared to the other tasks in that process. The task-sequence, observed through the recorded videos, are translated into Gantt Charts (GC) and Person-machine charts (PMC) to visualize the entire process. The recorded videos were analyzed on the computers at the department of Industrial & Systems Engineering of the University of Tennessee.

3.4.2 Management input

Since, every workstation has different processes, the precedence of tasks also differs accordingly. These differences may or may not be captured in the recorded videos. Additionally, the videos might not give information about company policies that need to be followed during manufacturing; for example, policies based on compliance or safety. There might be a specified task-sequence which need to be followed according to these policies.

To avoid missing out on information that cannot be captured on the video cameras, advice from the management can help fill such gaps. Managers or supervisors assigned at the workstation may help verify the correctness of the relative position of tasks.

3.4.3 Precedence table

A Precedence Table (PT) is a consolidated way to represent data related to task processing time, task-sequence and entire CT of the process. Precedence is a concept of scheduling used to define the relative position of each task as compared to all other tasks present at the workstation. Therefore, the requirement that ‘task i has precedence over task j ’, mandates

that task i must begin before task j in every scheduling scenario. For simplicity, task j can only begin after task i is finished. An example precedence table, provided in Table 3.1, shows precedence relationships between tasks at a workstation. This is represented by the values ‘100’, ‘0’ or ‘-100’ which are referred to as ‘Precedence factor f_p ’ in table 3.1.

Table 3.1: Jobs i on the axis, where $i = 1:5$

Tasks	1	2	3	4	5	Task Processing Time
1	0	100	100	100	100	6
2	-100	0	0	100	100	4
3	-100	0	0	0	0	10
4	-100	-100	0	0	100	8
5	-100	-100	0	0	0	7

The key points of the PT shown as an example in table 3.1 (i =row, j =column) are:

- The value of the cell represented by f_p defines the precedence of a task as compared to every other task present at the workstation. If $f_p = 0$, then the task in a row allows the task in a column to exist in parallel. If the $f_p = 100$, the task in a row has precedence over the task in a column.
- Numbers in the cells of the first column and first row, starting from the cell(0,0), represent tasks
- The “task processing time” column represents the duration of each task when it is assigned to a single employee
- All diagonal values of f_p are 0 since each task cannot have precedence over itself
- f_p value for (i,j) cell is -100 which means that task j is preceded by task i

3.4.4 Precedence scheduling

After data collection is complete, PBS generates a schedule based on the PT. Before aligning any task, all tasks are kept at $t = 0$ units of time. Each task extends from $t = 0$ units to their respective task processing times. An example is shown in Figure 3.2 for a process with five tasks.

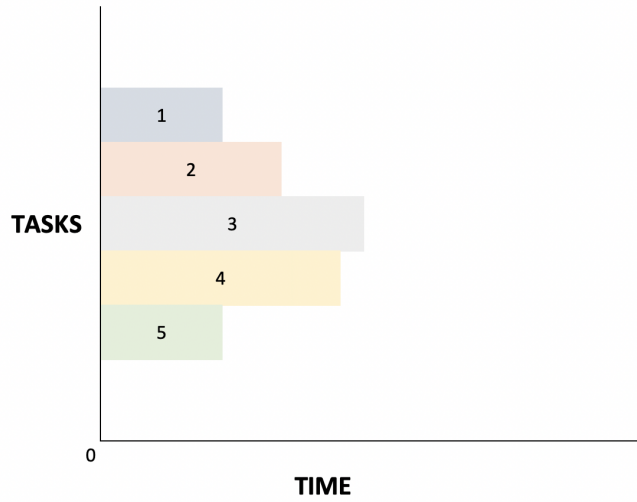


Figure 3.2: Tasks at $t = 0$ units before alignment on a GC

The realignment step shifts each task based on its f_p values as compared to the rest of the tasks. If the task at hand has precedence over the subsequent task, then the subsequent task's start time will shift to the right of the end time of the task at hand. Realigning tasks happens iteratively to guarantee that all precedence conditions have been satisfied.

A condition in the PBS algorithm is required to stop any further iterations of comparisons among the tasks. This condition simply compares the position of each task in the schedule generated in the current iteration with the one generated in the previous iteration. If the positions of these tasks are the same in both schedules, the iterations are stopped and the overall CT reduces in the latest schedule formed by PBS.

3.5 Zone-based scheduling

Zone-based Scheduling (ZBS) uses the schedule made by PBS to further reduce the cycle time. This is performed by using the following steps in the ZBS:

1. Assign employees to zones by sharing tasks
2. Commence a task based on employee availability
3. Verify whether or not the precedence is obeyed

3.5.1 Overlapping task

The objective of this step is to further reduce the CT based on the potential of scheduling tasks in parallel as defined in the PT. This can only happen when employees are available to work on the overlapping tasks.

Consider an example in Figure 3.4 with overlapping tasks A , B , and C at a workstation with three employees. The amount of time this overlap exists is called an “overlap period”. All three employees available at that workstation are engaged in these active-tasks during that overlap period. As a consequence, this overlap does not allow a new task to begin until an employee is available after any of these three tasks ends.

While assigning employees, a constraint called the “overlap rule” is developed. The overlap rule does not let the number of active-tasks exceed the number of employees assigned at the workstation during an overlap period. The decision of employee assignment depends on a correct evaluation of the overlap periods and the number of active-tasks at any instance during that overlap period.

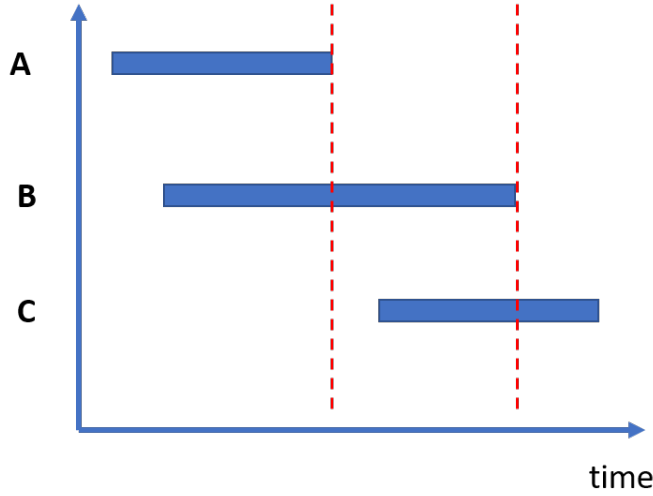


Figure 3.3: Example of an incorrect overlap rule violation inferred using a task-based perspective

It can be achieved by taking a task and observing the position of another task relative to that task at hand. Consider the example with three employees. Using this approach, if there are more than three active-tasks during that overlap, the overlap rule is breached. But with this constraint, there might be a possibility of rejecting a feasible task-sequence. Figure 3.3 portrays such a possibility of task-sequence which is feasible but may get rejected. If the task at hand is task B, then the overlap period is considered for all three tasks, i.e. A,B, and C. But in reality, only the job pairs (A, B) and (B, C) are overlapping. This way, if there were two employees instead of three, the overlap rule would not allow this possibility to occur. This way the employee assigned to task A might not be able to work on task C even after being available on task A's completion. Hence, there is a necessity to define overlaps. This case failure can be addressed by a modified approach to overlap period evaluation.

The resolution to the failure is by using the “overlap of overlap-periods”. The first step to achieve this is to find the first overlap by considering only two active-tasks. Figure 3.4 shows three overlaps. The hatched area indicates the merger of overlap periods between tasks A, B, and C. Task pairs (A,B), (B,C) and (A,C) form 3 different overlaps. Let OP_1 , OP_2 , and OP_3 be overlap periods of task pairs (A,B), (B,C) and (A,C) respectively. The outcome of

this example is that periods $[OP_1, OP_2, OP_3]$ merge into a single overlap. This overlapping of overlaps is shown in green which happens to be OP_3 . The number of active-tasks in overlap OP_3 is 3. This final overlap helps to avoid the redundancies in employee assignments. Also, this will ensure that the overlap between task A and task B and that between task B and task A, will register as a single entry.

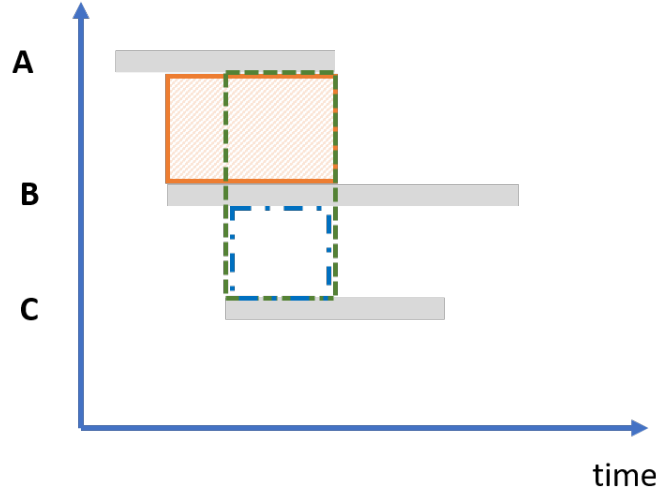


Figure 3.4: Example of a merger between overlap periods. The dotted and solid rectangular outlines indicate overlap periods. The hatched area indicates the merged overlap period, in which jobs A,B, and C are active.

Overlap of overlaps is iteratively computed and resolved until no further overlapping is possible. At the time of forming overlaps, the algorithm checks whether the number of active-tasks in any overlap period merger is less than K employees. If this condition is violated, then the start time of the last task during the overlap period shifts to the right of the end time of its previous task.

3.5.2 Precedence adjustments

This is the second step in ZBS and performs the same function performed by PBS, i.e. realigning the tasks based on precedence.

The inclusion of this step examines the number of active-tasks in an overlap period and shifts those tasks which; 1.) are preceded by other tasks having precedence over them, and 2.) do not have available employees which could be assigned to it. Violations for the overlap rule are checked after shifting the tasks. After this realignment, tasks may be redirected to follow the precedence constraints to avoid any violations based on precedence.

3.5.3 Zoning and employee assignment

In this step of ZBS, the objective is to assign the appropriate number of employees to each task and generate the final schedule of the process. Certain rules were developed to govern the assignment of employees. The following rules need to be obeyed when this step assigns employees to the tasks: 1.) Each task is allowed to have multiple employees assigned to it., 2.) If there is only one active-task, all employees at a workstation are added to that task, 3.) If there are multiple active-tasks, one employee is added to each task first, 4.) If employees are not available for an active-task, this task is shifted further right in the GC and does not begin until an employee is free, and 5.) The number of active-tasks in an overlap period should not exceed the number of employees assigned to that workstation.

Revisiting equation 3.1, t_j is the time taken by a single employee to complete task j . If there are multiple employees assigned to task j , its revised task processing time will be $\frac{t_j}{k}$ where k employees ($k > 1$) are assigned to task j . Since the revised task processing time is less than the previous task processing time, the overall CT of the process reduces.

The assignment is dependent on the assumption number 3 from section 3.3 which allows all tasks are capable of being divided into zones. When k employees share a task j , it is physically divided into k number of zones with each zone having no more than one employee assigned to it. These employees are cross-functional and equally skilled in performing all tasks at the workstation. Hence, each employee takes the same time to perform a shared task.

With the assignment of the required number of employees to the tasks, the final schedule of the process is complete. The final schedule only assigns the appropriate number of employees to the tasks and zones the tasks based on the condition. The assignment of a specific employee(s) per task does not happen yet as it is performed in section 3.6.

3.6 Workload balancing optimization

The next step, after PBS and ZBS, is the evaluation of the amount of work each employee is assigned. For this step, optimization using binary-integer linear programming is designed. The reason behind choosing optimization is to weigh the feasible possibilities of assignment of tasks to the employees to achieve the most equitable distribution of employee workload. The output obtained from PBS and ZBS method are 1. the number of employee(s) assigned to each task, and 2. revised task processing time for each task. The notation of the optimization model is presented below:

Sets

N Set of tasks

K Set of employees

Parameters

L A relatively big number

t_j $j \in N$ task processing time of task j

p_j $j \in N$ Number of available employees for task j

y_{jk} $j, k \in N$ 1 if task j has precedence over task k ; 0 otherwise

Binary Variables

X_{ij} $i \in K, j \in N$ 1 if employee i is assigned to task j ; 0 otherwise

Objective Function

$$\text{Minimize } \frac{1}{K} \sum_{i \in K} \left(\frac{1}{K} \sum_{i \in K} \sum_{j \in N} X_{ij} t_j - \sum_{j \in N} X_{ij} t_j \right)^2 \quad (3.2)$$

Constraints

$$\sum_{i \in K} X_{ij} = p_j \quad \forall j \in N \quad (3.3)$$

$$X_{ij} + X_{il} \leq 1 + L(1 - y_{jl}) \quad \forall i \in K, \quad \forall j, l \in N, l \neq j \quad (3.4)$$

Constraint 3.3 is to make sure that the number of employees assigned to a task should not be more than the available employees. Constraint 3.4 is written to make sure that if two tasks overlap, then an employee is assigned to one of them. This resolves the issue of assigning one employee to more than one active-tasks. The workload balancing is achieved by the objective function 3.2 which minimizes the variation between the workload of each employee. This is performed by first calculating the workload of each employee and later minimizing the difference between the average workload per employee and workload per employee.

Chapter 4

Case study

The ZBM methodology has been piloted at a facility of Company ABC. The workstations at this facility were studied using tools like time study, person-machine charts and Gantt charts which are discussed by showing how data collection was performed at the chosen workstation. Later, the results were obtained from the application of ZBM at the workstation observed at this facility.

4.1 About Company ABC

Company ABC is ISO 14001 Certified for green building standards [ISO]. They purchase building materials in bulk which are measured to exact specifications for customers' satisfaction. Necessary materials for construction, such as lumber, insulation, OSB, floor decking and flooring finishes, are a few materials that are used in production. Facilities of this company usually take six-seven days to manufacture a house while a few of the fastest facilities take two days depending upon the level of customization involved in the design. The facility of the Company ABC considered for this case study is capable of producing four houses a day.

4.2 Details of the facility

The layout of the test facility is shown in figure 4.1 which consists of 12 workstations. This facility provides several customization options for each product. Regardless of the product, the task processing time for the same task remains unchanged across products. For instance, the task processing time of placing a wooden slab on the floor is the same for all products in which it is performed.

To manufacture a product at this workstation, the employees were provided with the list of required tasks. The employees were not cross-trained for these tasks required at the workstation. However, no guidelines on obeying any task-sequence were provided to the employees. This led to an unregulated decision-making environment which distributed the workload unequally. This is revealed in the observations through time study. Another factor that contributed to unequal workload was the lack of sharing of tasks among the employees. For example, employees at workstation 8 and workstation 10 shared a task of painting different sections of the house, with 2 and 4 employees sharing the workload respectively.

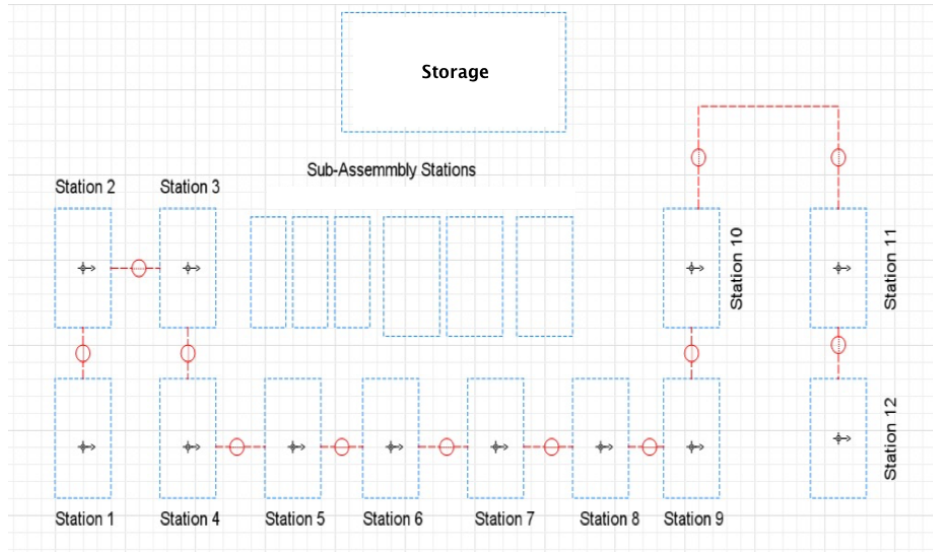


Figure 4.1: The layout of the facility at Company ABC's facility

4.3 Floor-jig case

4.3.1 Precedence-based scheduling

Data collection

The following points were taken into consideration while selecting cameras for video recording: 1.) Cameras with the capability of capturing at least 30 frames per second, 2.) The capacity of at least 32 gigabytes of memory, and 3.) Auto-start and stop capability.

Observations were made through the video recordings generated by the cameras to track actions performed by all employee at the workstation. The pictures shown in figure 4.2 show the workstation known as “Floor Jig” workstation from multiple angles. Each employee’s hat was uniquely colored to establish a clear distinction among all employees assigned at the workstation. This is called “man-marking” as mentioned in table 4.1. Table 4.1 is constructed to show the guidelines followed in the data collection process. The left column has guidelines related to both task-related and employee-related data collection. The right column shows the respective tools used to perform the data collection.



Figure 4.2: Recordings of Floor Jig through multiple angles

Table 4.1: Guideline to collect time study data

Work Activities - Data Collection	Tools
<i>I. Employee-related data collection</i> 1. Record the workstation using cameras mentioned 2. Select an employee from the recorded video data 3. Track all tasks performed by the employee 4. Register the time for each task with a detailed description 5. Track and register until the final task by the employee is performed before the manufacturing process is over 6. Repeat from step 2. for the next employee	• Project selection criteria • Video recording • Man-marking • Person-machine chart (figure 4.3)
<i>II. Task-related data collection</i> 1. Measure the time required to perform each task 2. Note if the task can be performed by multiple employees at the same time, i.e. if employees can share a task 3. Identify a task as either value-added (VA) or non-value-added (NVA) 4. List out all possible tasks performed by different employees with their respective processing time	• Time study • Gantt chart (4.4) • Precedence Table (??)

The recordings were later transcribed in a format of the person-machine chart (PMC). The Floor Jig had three employees working for a process, hence there were three PMCs developed for that workstation as each employee required one PMC. Each PMC shows actions performed by an employee until the end of the process. Shared tasks can be identified if they are mentioned in multiple PMCs which were mentioned in the description column of each PMC as shown in figure 4.3.

A	B	C	D	E	F	G	H	I	J	K	L
1			Diagram:	PROCESS MM CHART						Date:	1/25/2018
2			Description:	CLAYTON HOMES						Diagram No.:	1
3			Operator Name:	Lucas Collaps						Station:	1
4			Time In:	11:28						Time Out:	12:57
5											
6	From	To	Time (Minutes)	Symbols					VALUE ADDED/ NON VALUE ADDED	Description	Observations (if any)
7				○	→	□	◐	▽			
8			0:00:00								
9	11:29:00	11:29:26	0:00:26		x				NV	move from station to inventory	
10	11:29:26	11:29:31	0:00:05						NV	spends time in inventory	
11	11:29:31	11:29:39	0:00:08		x				NV	collects the material and comes back to station	
12	11:29:39	11:30:44	0:01:05	x					V	adjusts , cut , joints the wooden logs	
13	11:30:44	11:31:49	0:01:05		x				NV	goes to the other end of station	
14	11:31:49	11:33:09	0:01:20	x					V	marking on the wooden logs	
15	11:33:09	11:33:20	0:00:11						NV	wait for the other employee	
16	11:33:20	11:35:58	0:02:38	x					V	marking the wooden logs	
17	11:35:58	11:36:47	0:00:49		x				NV	standing	
18	11:36:47	11:37:03	0:00:16	x					V	adjusting the wooden logs	
19	11:37:03	11:37:07	0:00:04		x				NV	moves towards the station	
20	11:37:07	11:37:29	0:00:22	x					V	covers the sheet around the logs and starts punching	

Figure 4.3: Snapshot of PMC which shows employee engagement at workstations for the case discussed

The columns named “From” and “To” have the start time and end time of each task respectively. The column named “Time (Minutes)” shows the difference in start time and end time of each task. This provides the input for calculating the task processing time. The symbols used are for different kinds of tasks performed by the employees. The first column, under symbols, is for operations. Only these tasks are taken into account as they fall under value-added tasks. Column named ‘Description’ describes the task in words.

It represents the tasks at a workstation and the employees assigned to the tasks. GC provides visualization of the work being performed at any given period at a workstation. The horizontal axis on the GC represents time, and the rows denote all the tasks present at that workstation. This allows the reader to observe: 1.) Duration of each task, 2.) The number and identity of employees hired to perform those tasks, 3.) Cumulative value of all task processing time, i.e. cycle time of the process at a workstation, and 4.) Availability of employees at any given point of time.

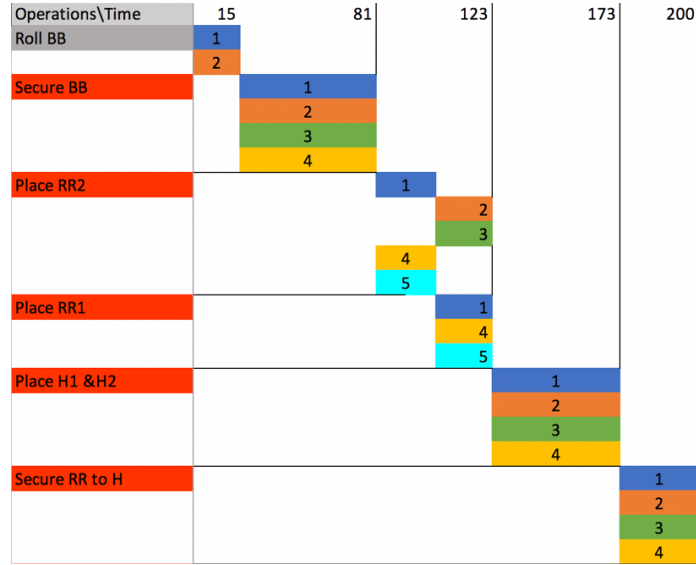


Figure 4.4: An example of a Gantt Chart including 5 employees

The different colors are a representation of the number of employees working on that task. This helps in visualizing the utilization of employees at the end of the process. The information displayed in the GCs is further used in generating precedence tables.

Floor-jig case represents a complete process at Floor Jig for a different model of a product manufactured at the facility. Tasks observed for this product were less sequential and thus having less $f_p = 100$ values or more of $f_p = 0$ values in the PT. This way, the task-sequence potentially allows more overlaps of tasks. Consequentially, it becomes challenging to assign employees in this case. The process observed at Floor Jig is shown in figure 4.5. This process took 74 minutes of CT with the employee assignment shown in table 4.2.

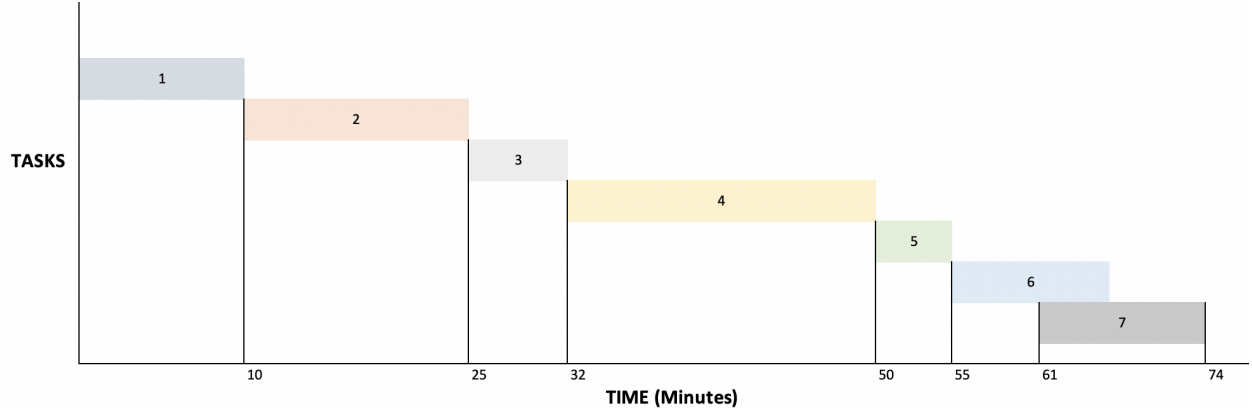


Figure 4.5: Floor-jig case - Gantt Chart of tasks at Floor Jig before application of ZBM

Employee assignment observed before the application of ZBM is shown in table 4.2.

Table 4.2: Employee assignment before ZBM

Employees	Workload (time units)	Tasks						
		1	2	3	4	5	6	7
A	43	✓	✓		✓			
B	17					✓	✓	
C	20			✓				✓

Management input

The only compliance required by the facility was to have task 4 complete before tasks 5, 6, and 7. Although task 4 and task 5 could have been performed simultaneously, task 5 was required to begin after task 4 due to company policy. This became a constraint that was strictly required and is reflected in the task-sequence shown in figure 4.7.

Precedence table

A PT generated for this case is shown in table 4.3. The task processing time for each task and its f_p values are listed. The PT shown in table 4.3 shows the relation between all tasks.

For example, $f_p = 0$ in the row of task 1 for columns of tasks 4 and 6. This means 1 does not have precedence over tasks 4 and 6.

Table 4.3: Floor-jig case of input

Tasks	1	2	3	4	5	6	7	Processing Time
1	0	100	100	0	100	0	100	10
2	-100	0	100	100	0	100	100	15
3	-100	-100	0	0	100	100	100	7
4	0	-100	0	0	100	0	100	18
5	-100	0	-100	-100	0	100	0	5
6	0	-100	-100	0	-100	0	0	12
7	-100	-100	-100	-100	0	0	0	13

Precedence scheduling

The Floor-jig case has three employees working at the Floor Jig workstation. When the tasks are lined up consecutively, without having any overlaps among the tasks, the CT of the process adds up to 80 minutes. The alignments are done using PBS which provides a substantial difference in cycle time by dropping the CT from 74 minutes to 60 minutes. The difference between the observed process and the process after PBS was implemented is 14 minutes. The schedule computed after PBS is shown in figure [4.6](#).

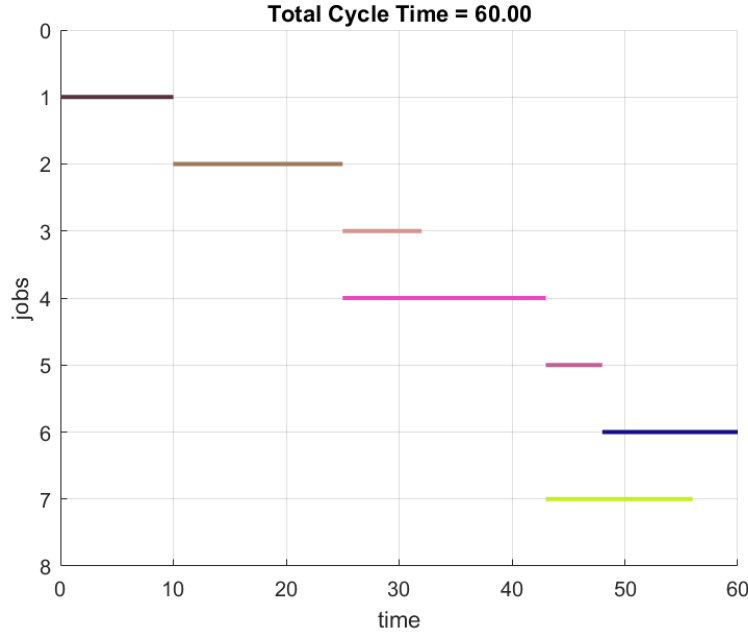


Figure 4.6: Floor-jig case - output of precedence scheduling. The resultant cycle time is 60 minutes.

4.3.2 Zone-based scheduling

Overlapping Task

As observed in figure 4.6, three overlaps exist in parallel: tasks [3, 4] overlap, tasks [5, 7] overlap, and tasks [6, 7] overlap. As mentioned earlier 3.5 under ‘Task Overlapping’, the task group [5, 6, 7] is not considered to be a single group, as expected. All three overlaps have at most three active tasks.

Precedence adjustments

The number of active tasks, during any overlap, does not exceed the number of employees assigned at the workstation, which does not violate the overlap rule. This is why the task-sequence showed figure 4.6 and figure 4.7 remain the same after overlapping tasks were found. If the overlap rule would have been violated, then all tasks after the K^{th} task in an overlap would have shifted to the right with the use of precedence factors.

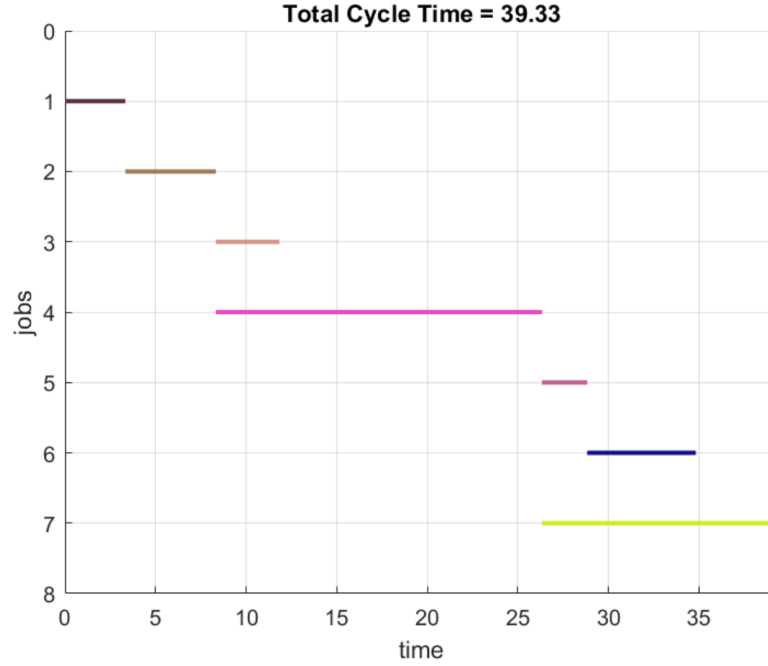


Figure 4.7: Floor-jig case - output of Zone-based scheduling. The resultant cycle time is 39.33 minutes.

Zoning and employee assignment

The challenge for the algorithm is to assign employees specifically during the overlap of the task overlap groups $[5, 7]$ and $[6, 7]$. The algorithm assigns the employees to $[5, 6]$ first and then assigns them to the $[6, 7]$. The test is to not assign different employee(s) to the active-tasks 6 and 7 once employees are freed from task 5.

Only task 4 and task 7 are not zoned, while remaining tasks are divided either into two or three zones. At most three zones are made for any task since the number of the employees assigned to the workstation is $K = 3$.

Figure 4.7 shows a further reduction of cycle time in the process after precedence scheduling. The new cycle time is 39.33 minutes. The column 'K' in table 4.4 shows the resultant assignment of employees obtained at the end of the ZBS. The column 'Revised TPT' shows the revised values of the task processing time of each task.

Table 4.4: Floor-jig case: Results of zone-based scheduling and employee assignment. K denotes the number of assigned employees

Tasks	K	Start Time	End Time	Revised TPT	Old TPT
1	3	0.00	3.33	3.33	10.00
2	3	3.33	8.33	5.00	15.00
3	2	8.33	11.83	3.50	7.00
4	1	8.33	26.33	18.00	18.00
5	2	26.33	28.83	2.50	5.00
6	2	28.83	34.83	6.00	12.00
7	1	26.33	39.33	13.00	13.00

4.3.3 Workload balancing optimization

Objective function

The standard deviation (SD) of the workload among the employees, is a square root of the objective function 3.2. This is shown as equation 4.1. The lower the value of SD, the better the result in terms of workload distribution. The balance of workload is best achieved by the distribution of workload mention in table 4.5. The standard deviation using equation 4.1, is 6.05989.

$$SD_{Workload} = \sqrt{\frac{1}{K} \sum_{i \in K} \left(\frac{1}{K} \sum_{i \in K} \sum_{j \in N} X_{ij} t_j - \sum_{j \in N} X_{ij} t_j \right)^2} \quad (4.1)$$

Constraints

The constraint 3.3 does not assign more employees than the number of employees suggested by ZBS in table 4.4. The assignment based on this constraint is shown in table 4.5 with check marks representing the task assigned to the employee.

Table 4.5: Floor-jig case results of the employee assigned to minimize the variation of workload

Employees	Workload (time units)	Tasks						
		1	2	3	4	5	6	7
Emp1	34.83	✓	✓		✓	✓	✓	
Emp2	24.83	✓	✓	✓				✓
Emp3	20.33	✓	✓	✓		✓	✓	

The workload distribution of this case is shown in figure 4.8 is the most optimal solution obtained for the schedule shown in figure 4.7. The discrepancy in the workload among the employees is observed in figure 4.8. It shows Employee 1 is working the most and Employee 3 is working the least.

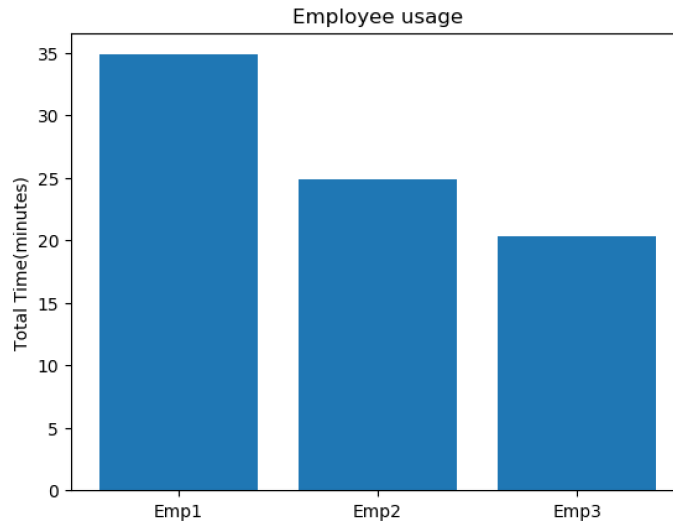


Figure 4.8: Employee time contribution ($K = 3$)

4.4 Job-Task Analysis-II development

4.4.1 Job-Task Analysis by Company ABC

Job-Task Analysis (JTA) is a process developed by Company ABC to identify the responsibilities of all employees to calculate their workload. Responsibilities are a list of tasks that need to be performed by the employees to finish a product at the workstation. JTA identifies each task and its task processing time. Tasks inside the responsibilities are identified based on employee interviews and the task processing time is determined by direct observation.

The identified responsibilities and their tasks, for the presented case study, are shown in figure 4.9. The figure shows the JTA for each employee's responsibility, at the Floor Jig, listed along with the total time taken to finish the responsibility. An acronym is used for tasks to specify the responsibility it falls under. For example, task R1T1 belongs to Responsibility 1, whereas R3T7 belongs to Responsibility 3.

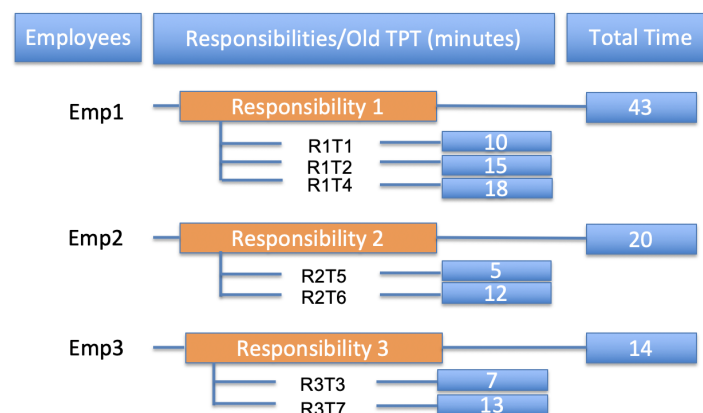


Figure 4.9: Job-Task Analysis of three employees and their respective responsibilities

In figure 4.9, each task was uniquely assigned to an employee which means the employees did not share tasks. The longest working employee is 'Emp1' as his responsibility lasted for a total time of 43 minutes.

4.4.2 Job-Task Analysis-II

Job-task Analysis-II (JTA-II), shown in figure 4.10, is a revised version of the JTA that was developed after the application of the ZBM at a workstation. Unlike JTA, the purpose of JTA-II is more than the identification of the responsibilities of the employees. Responsibilities mentioned in JTA-II can be used as the basis for developing future schedules. Responsibilities mentioned in JTA-II can be used as the basis for developing future schedules.

JTA-II displays revised responsibilities of employees based on scheduling and employee assignment in ZBM. These revised responsibilities set expectations for their respective employees to finish the process. For instance, in figure 4.10 ‘Emp1’ is expected to perform all five tasks under his/her ‘Responsibility 1’.

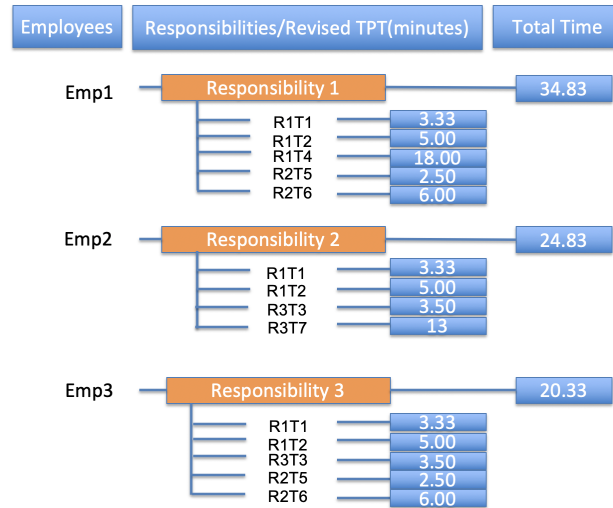


Figure 4.10: Revised Job-Task Analysis of three employees and their respective responsibilities

JTA-II mentions a single task under multiple responsibilities, as this task is shared by multiple employees as a result of Workload Balancing. A team is formed of the employees who share tasks. The set of employees assigned to share a task is referred to as a “team” for that task. In figure 4.10, as a result of task ‘R1T1’ being shared by a team of three employees, the task is distributed in three zones. This results in an equal distribution of tasks per responsibility. Although there were more tasks added per responsibility, the total time taken by employees to perform their responsibilities became more balanced.

Chapter 5

Discussion and future work

The central idea of the current model is that of zoning, in which a workstation for each task is divided into smaller areas called zones and one employee is assigned to each zone. This reduces the workload per employee and improves the overall cycle time because of the division of labor. ZBM is formulated and implemented as an algorithm; the only input needed is a precedence table for each process and its tasks. This allows the manager to use the system, even if they are non-experts in work scheduling and assignment. Work assignments are balanced using optimization, to guarantee that work is equitably distributed and no employee is loaded with additional work under the guise of improved cycle time. The approach is tested using a case study conducted by Company ABC. The algorithm works as designed, and cycle time reduction exceeding 50% of the original value is observed in both cases in the study.

Sections [5.1](#) and [5.2](#) provide some suggestions which would improve the current version of ZBM and make it more versatile.

5.1 Concept of super-zones

5.1.1 What is a super-zone?

A “super-zone” is a collection of all the zones of a single task. In figure [5.1](#), task A is divided into three rectangular zones. The biggest square is the super-zone for task A that contains

all zones of task A. Each zoned task has a super-zone for itself and hence there are five different super-zones in the figure below.

In a practical setting, each task requires specific skills from employees. Under the concept of super-zone, the employees will be assigned to the super-zones according to the skills they possess. A team of employees with similar skill-sets in a super-zone may not be required to perform other tasks present in other super-zones. The presence of multiple teams increases the availability of employees to perform parallel tasks.

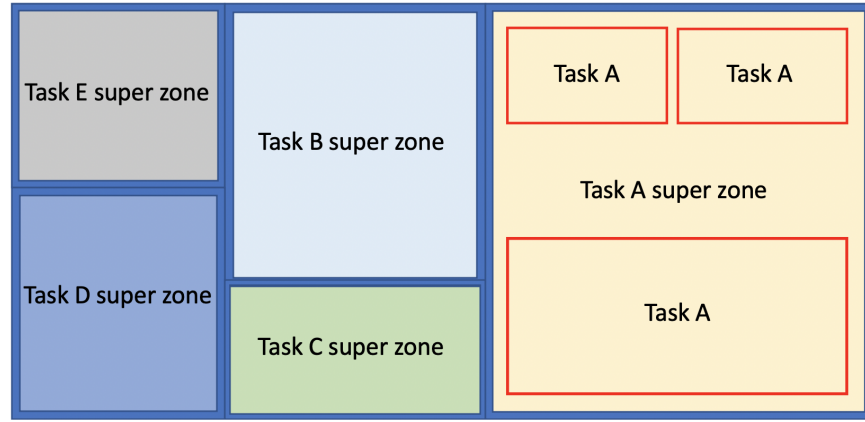


Figure 5.1: Employee time contribution ($K = 3$)

5.1.2 Benefits of super-zone

Better division of work

A shortcoming of this version of ZBM is that it can only be used at workstations with cross-functional employees. Hence, super-zones are needed to overcome this shortcoming in workstations where employees are not cross-functional. The methodology of super-zoning the workstation will consider the skill-set of certain employees and assign them to super-zones accordingly.

Material handling

The teams, working in several super-zones, require different materials. These specific materials need to be brought near the super-zone where it would be utilized. In this way employees do not need to fetch materials from a distant storage unit, which is a non value-added activity. The benefit of doing this is reduced time to fetch materials. Hence super-zone, as a potential inclusion to ZBM, reduces this non value-added activity and increases the overlap of tasks in order to reduce the cycle time without the need of cross-functional employees.

5.1.3 Inclusion for algorithm in super-zone

The scheduling algorithm using super-zones will first assign a team of multiple employees to a super-zone. Later, the individual employees on that team form zones inside that super-zone based on the number of employees in the team. In figure 5.1, the team assigned to task A has three employees and hence the super-zone of task A contains three zones. The idea of inclusion of teams in scheduling using a super-zone represents a more realistic manufacturing environment where not all employees are cross-functional.

5.2 Other suggestions

There are several promising directions for this work to take. In addition to absolute time demands, work can be stressful on production lines because of insufficient or poorly timed rest periods. Consideration of rest time allowance can be built into the algorithm, to supplement the equal workload feature. The current work assumes completely cross-functional teams. This assumption can be relaxed by factoring in employee-specific skills into the allocation of tasks. The optimization and heuristic elements of the presented approach function as independent software. The next version of the software will provide a seamless end-to-end experience for the user, in which the precedence table is the only software input and interaction required from the user. It will be interesting to test the software and the algorithm with more complex cases, in which work may span longer periods which include scheduled

events such as shift changes. Finally, it is pertinent to examine zoning from the ergonomic standpoint, to better define the nature of zoned tasks.

Bibliography

- [1] Afzalirad, M. and Rezaeian, J. (2016). Resource-constrained unrelated parallel machine scheduling problem with sequence dependent setup times, precedence constraints and machine eligibility restrictions. *Computers & Industrial Engineering*, 98:40 – 52. [8](#), [11](#), [13](#), [18](#)
- [2] Al-Araidah, O., Momani, A., Khasawneh, M., and Momani, M. (2010). Lead-time reduction utilizing lean tools applied to healthcare: The inpatient pharmacy at a local hospital. *Journal for healthcare quality : official publication of the National Association for Healthcare Quality*, 32:59–66. [8](#), [13](#)
- [3] Bachouch, R. B., Guinet, A., and Hajri-Gabouj, S. (2010). An optimization model for task assignment in home health care. In *2010 IEEE Workshop on Health Care Management (WHCM)*, pages 1–6. [15](#), [16](#), [18](#)
- [4] Battaïa, O., Delorme, X., Dolgui, A., Hagemann, J., Horlemann, A., Kovalev, S., and Malyutin, S. (2015). Workforce minimization for a mixed-model assembly line in the automotive industry. *International Journal of Production Economics*, 170:489–500. [5](#), [15](#), [16](#)
- [5] Çaliş, B. and Bulkan, S. (2015). A research survey: review of ai solution strategies of job shop scheduling problem. *Journal of Intelligent Manufacturing*, 26(5):961–973. [10](#)
- [6] Chaudhry, I. A. and Khan, A. A. (2016). A research survey: review of flexible job shop scheduling techniques. *International Transactions in Operational Research*, 23(3):551–591. [6](#), [10](#)
- [7] Cheng, R., Gen, M., and Tsujimura, Y. (1996). A tutorial survey of job-shop scheduling problems using genetic algorithms—i. representation. *Computers & industrial engineering*, 30(4):983–997. [5](#)
- [8] Conti, R., Angelis, J., Cooper, C., Faragher, B., and Gill, C. (2006). The effects of lean production on worker job stress. *International Journal of Operations Production Management*, 26(9):1013–1038. [1](#)

- [9] Costa, A., Cappadonna, F., and Fichera, S. (2013). A hybrid genetic algorithm for job sequencing and worker allocation in parallel unrelated machines with sequence-dependent setup times. *The International Journal of Advanced Manufacturing Technology*, 69. [9](#), [13](#), [18](#)
- [10] Craig, G., Wellener, P., Dollar, B., Manolian, H., Monck, L., and Mooutray, C. (2018). 2018 skills gap in manufacturing study. *Deloitte Insights*. [1](#)
- [11] Demir, Y. and Isleyen, S. K. (2013). Evaluation of mathematical models for flexible job-shop scheduling problems. *Applied Mathematical Modelling*, 37(3):977–988. [11](#), [13](#), [18](#)
- [12] Dolgui, A., Kovalev, S., Kovalyov, M. Y., Malyutin, S., and Soukhal, A. (2018). Optimal workforce assignment to operations of a paced assembly line. *European Journal of Operational Research*, 264(1):200 – 211. [10](#)
- [13] Fattahi, P., Jolai, F., and Arkat, J. (2009). Flexible job shop scheduling with overlapping in operations. *Applied Mathematical Modelling*, 33(7):3076 – 3087. [10](#), [12](#), [18](#)
- [14] Gupta, J. N. and Stafford Jr, E. F. (2006). Flowshop scheduling research after five decades. *European Journal of Operational Research*, 169(3):699–711. [10](#)
- [15] Haller, M., Peikert, A., and Thoma, J. (2003). Cycle time management during production ramp-up. *Robotics and Computer-Integrated Manufacturing*, 19(1-2):183–188. [1](#)
- [ISO] ISO. ISO, Geneva, Switzerland. [32](#)
- [17] Jain, A. S. and Meeran, S. (1999). Deterministic job-shop scheduling: Past, present and future. *European journal of operational research*, 113(2):390–434. [10](#)
- [18] Khalife, M., Abbasi, B., and Abadi, A. (2010). A simulated annealing algorithm for multi objective flexible job shop scheduling with overlapping in operations. *J. Ind. Eng.*, 5:17–28. [12](#), [13](#), [18](#)

- [19] Kwon, B. and Lee, G. M. (2015). Spatial scheduling for large assembly blocks in shipbuilding. *Computers Industrial Engineering*, 89:203 – 212. Maritime logistics and transportation intelligence. [14](#), [18](#)
- [20] Lee, S., Moon, I., Bae, H., and Kim, J. (2012). Flexible job-shop scheduling problems with ‘and’/‘or’ precedence constraints. *International Journal of Production Research*, 50(7):1979–2001. [11](#)
- [21] Lei, D. (2008). Multi-objective production scheduling: a survey. *The International Journal of Advanced Manufacturing Technology*, 43(9):926. [12](#)
- [22] Lerman, E., S., Eskin, J., E., Flower, C., D., George, R., E., Gerson, R., B., Hartenbaum, R., N., Hursh, R., S., and Moore-Ede, R., M. (2012). Fatigue risk management in the workplace. *Journal of Occupational and Environmental Medicine*, 54(2):231–258. [2](#)
- [23] Li, J. (2013). Continuous improvement at toyota manufacturing plant: applications of production systems engineering methods. *International Journal of Production Research*, 51(23-24):7235–7249. [1](#)
- [24] Lian, J., Liu, C., Li, W., and Yin, Y. (2018). A multi-skilled worker assignment problem in seru production systems considering the worker heterogeneity. *Computers & Industrial Engineering*, 118:366–382. [15](#), [18](#)
- [25] MacBryde, J., Radnor, Z., Taj, S., and Berro, L. (2006). Application of constrained management and lean manufacturing in developing best practices for productivity improvement in an auto-assembly plant. *International Journal of Productivity and Performance Management*, 55(3/4):332–345. [1](#)
- [26] Mutlu, S., Benneyan, J., Terrell, J., Jordan, V., and Turkcan, A. (2015). A co-availability scheduling model for coordinating multi-disciplinary care teams. *International Journal of Production Research*, 53(24):7226–7237. [9](#)
- [27] Nie, H., Liu, B., Xie, P., Liu, Z., and Yang, H. (2014). A cuckoo search algorithm for scheduling multi-skilled workforce. *Journal of Networks*, 9. [9](#), [10](#), [11](#), [12](#), [18](#)

- [28] Panwar, A. and Jha, K. N. (2019). A many-objective optimization model for construction scheduling. *Construction Management and Economics*, 37(12):727–739. [8](#), [10](#), [13](#), [18](#)
- [29] Pastor, R., Andrés, C., Duran, A., and Pérez, M. (2002). Tabu search algorithms for an industrial multi-product and multi-objective assembly line balancing problem, with reduction of the task dispersion. *Journal of the Operational Research Society*, 53(12):1317–1323. [15](#)
- [30] Pattanaik, L. and Sharma, B. (2009). Implementing lean manufacturing with cellular layout: a case study. *The International Journal of Advanced Manufacturing Technology*, 42(7-8):772–779. [1](#)
- [31] Rajan, D. (2018). Negative impacts of heavy workload: A comparative study among sanitary workers. *Sociology International Journal*, 2(6):465–447. [1](#)
- [32] Rajendran, C. and Ziegler, H. (2004). Ant-colony algorithms for permutation flowshop scheduling to minimize makespan/total flowtime of jobs. *European Journal of Operational Research*, 155(2):426 – 438. Financial Risk in Open Economies. [8](#), [10](#), [12](#)
- [33] Ronnachai, S. and Chutima, P. (2010). The impact of walking time on u-shaped assembly line worker allocation problems. *Engineering Journal*, 14. [1](#), [5](#), [9](#), [11](#), [13](#), [15](#), [18](#)
- [34] Saidi-Mehrabad, M. and Fattahi, P. (2007). Flexible job shop scheduling with tabu search algorithms. *The International Journal of Advanced Manufacturing Technology*, 32:563–570. [8](#), [12](#)
- [35] Slomp, J., Bokhorst, J. A., and Molleman, E. (2005). Cross-training in a cellular manufacturing environment. *Computers & Industrial Engineering*, 48(3):609 – 624. GroupTechnology/Cellular Manufacturing. [15](#), [16](#)
- [36] Tse, E., Histon, J., Scott, S. D., and Greenberg, S. (2004). Avoiding interference: How people use spatial separation and partitioning in sdg workspaces. [14](#)

Vita

Abhay Bajpai, son of Anita Bajpai and Rajendra Bajpai, was born in Kanpur, India, on June 22, 1993. He received a B.Tech degree in Mechanical Engineering program from SRM University in India. He enrolled in the Master of Science program in Industrial and Systems Engineering at the University of Tennessee, Knoxville in 2017 and served as both, Graduate Research Assistant and Graduate Teaching Assistant during his program. He finished his Masters of Science degree with a minor in statistics in May 2020.